Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 81 (2019) 707–712

www.elsevier.com/locate/procedia

52nd CIRP Conference on Manufacturing Systems

Redistributed manufacturing of spare parts: an agent-based modelling approach

Yousef Haddad, Konstantinos Salonitis*, Christos Emmanouilidis

Manufacturing Department, Cranfield Universit, Bedford, MK43 0AL, UK

* Corresponding author. Tel.: +44 (0)1234 758347; *E-mail address:* k.salonitis@cranfield.ac.uk

Abstract

Maintenance and repair activities from the perspective of OEMs are both considerable sources of revenue and expenses, particularly when part of a Product Service System (PSS). It is therefore necessary for an OEM that provides services bundled with products to ensure timely response without significant impact on cost. This paper proposes a make-to-order spare parts supply chain strategy through the adoption of Redistributed Manufacturing (RdM) where the supply chain is shortened and total cost is decreased. An agent-based model that portrays an OEM's response to repair a failed equipment is developed to exhibit the potential time and cost savings gained by OEMs.

Keywords: redistributed manufacturing; product service system; agent-based modelling

1. Introduction

In an increasingly competitive after-sales market, maintenance, repair and operations (MRO) activities constitute both a considerable source of revenue as well as a significant source of cost for original equipment manufacturers (OEMs). As OEMs strive for customers' satisfaction, key decisions have to be made to ensure timely and efficient MRO offerings while maintaining associated costs at an acceptable level. Timely and efficient MRO offerings would naturally require abundance of resources such as full inventory of spare parts, skilled technicians and transportation vehicles always available and in close proximity to demand points. This setting, however, entails high cost and low resource utilization which are difficult to justify and often prohibitively costly.

This paper proposes redistributed manufacturing (RdM) for the localized production of spare parts. RdM is an emerging manufacturing paradigm closely related to but distinct from ubiquitous manufacturing (UM); where the former revolves around decentralization of operations, resources and decision making, while the latter imply the integration of ubiquitous computing (UC) into production activities [1]. More specifically, RdM entails the shift from centralized mass production towards smaller geographically dispersed manufacturing facilities located in close proximity to the end user, and interconnected through information and communication technologies (ICT) and empowered by the latest manufacturing technologies [2,3] such as additive manufacturing (AM) [4].

In this paper, a simulation model is presented to assess the impact of RdM on the performance of MRO offerings from the perspective of OEMs. Two hypothetical scenarios are formulated on the basis of thorough investigation of case studies in the contemporary literature as well as reports from corporation that employ RdM in the production of spare parts. Both scenarios are then simulated from a bottom-up approach using agent-based modelling (ABM) to assess the impact of RdM on the total cost of MRO, machines downtime at the clients' premises and utilization of resources.

10.1016/j.procir.2019.03.180

2. Related Work

Multiple models and frameworks utilizing different techniques have been developed to aid in the decision making in the area of spare parts logistics [5,6]. However the area of on-demand distributed production of spare that is enabled by advanced ICT and manufacturing technologies is relatively new; with [7] being one of the first attempts in this field. Much of the work in this area has targeted the aerospace sector [8–12] due to the significant costs associated with the aircrafts' grounding. Other works, however, developed hypothesized supply chains independent of any specific sector to assess the impact of decentralization coupled with the production using AM technology on cost and carbon footprint [13], or departed from the aerospace sector [14] and investigated the impact of AM on different spare parts supply chains. Most of the reviewed papers assessed supply chains' performance on either cost of introducing and operating under RdM utilizing AM [8,9,11,13], or different inventory performance metrics [10,12,14], or both cost and carbon footprint [13].

Most of the research in this area found a significant advantage of on-demand production of spare parts using AM technologies. However, much of the contemporary research concludes that, with the current state of the existing AM technologies, the centralized setting of spare parts production is preferable to the distributed one [8,9,12].

The use of simulation modelling in the distributed production of spare parts is somehow limited. Only two of the reviewed [12,13] papers used simulation modelling, particularly system dynamics (SD) as the main tool of investigating the impact of distributed AM-enabled production of spare parts on supply chains. Two further papers [9,11] used simulation in a limited format; the authors used Monte Carlo simulation to complement scenario modelling in order to model inventory stock-outs. Two of the papers were qualitative in nature [8,14]; where techniques such as conceptual designs [8] and systems theory and dynamics capabilities [14] were used in this context. These studies, although lack the empirical side that could verify the findings, provide valuable insights into distributed spare parts production and the introduction of AM to the spare parts supply chains. One paper [10] used the supply chain reference model (SCOR) to quantitatively analyze the impact of the distributed production of spare parts on the inventory safety stock of a hypothetical supply chain.

This paper aims fill a part of the gap in the use of simulation modelling in the area of on-demand distributed production of spare parts. This research takes a bottom-up approach where individual heterogeneous entities and processes are modelled, and the systems' behavior emerges as a result of the interaction between different entities and entities with the environment they inhabit.

3. Generic repair process

Two models representing two hypothetical generic repair processes are formulated for this study. The generic repair process in this context refers to a high level representation of attending to a failed machine where the basic functions performed in most repair processes and the ones that are relevant to the objective of this research are included. The functions in the repair process are attending to the failed machine by a technician, inspection, ordering parts (if necessary), and repairing. Both models represent a fleet of machines of a similar model, all prone to failures, distributed over a given geographical area.

The two modelled scenarios are namely the Traditional scenario and the RdM one, both named after the production approach. It is worth mentioning here that the RdM scenario adopts hub setting i.e. not fully distributed; where the geographical area where the machines are placed is divided to zones, each containing a number of machines and has its own dedicated resources that are shared between the zone's constituents. The Traditional scenario consists of the same geographical area with the same machines and the same resources; which are all centralized.

The repair process in the Traditional scenario goes as follows: when a machine breaks down, a repair request is sent to the maintenance center where it joins a queue and is picked on a first-come-first-serve basis by the first available technician. The technician then travels to the failed machine, inspects it, and if no parts are required repairs it. If parts are needed, the technician orders the required parts from the warehouse, checks whether there are any other broken machines that need attending to, and then leaves either for a new job or back to the maintenance center. Meanwhile, if the technician decides that parts are needed for repair, the failed machine joins a queue for machines that are grounded until their respective parts arrive either from the central warehouse, or from the parts' respective suppliers in the event of a stock-out.

The repair process in the RdM scenario, which is presented in Fig. 1. below, is triggered with the breakdown of a machine. Once a machine breaks down, a repair request is sent to the machine's respective zone's maintenance center, which contains a local storage facility that contains all the parts in limited quantities and is replenished daily, and AM equipment for on-demand manufacturing. The repair request then joins a queue and is picked from the respective zone's technician on a first-come-first-serve basis. The zone's respective technician then heads to inspect the failed machine, if no parts are needed then the technician repairs the machine in the same visit. If parts are required, then the technician checks what parts could be manufactured on-demand in the corresponding local maintenance center, and sends a request to the zone's local maintenance center to produce these parts; the request to produce parts on-demand joins a queue and is picked once the first AM equipment becomes available. If the required parts cannot be produced on-demand then the technician checks the parts availability at the zone's maintenance center storage area, where if all the required parts are available they are reserved. If the required parts are not available at the local maintenance center storage facility, then the technician checks with the central warehouse. In case of stock-out at the central warehouse, the parts are requested from the respective suppliers(s). Based on the conceptual model, the next section presents the ABM simulation model.

4. Agent-based simulation model

Since the objective of modelling in this research is to improve the performance of a system through observing and evaluating what-if scenarios, simulation modelling has been chosen as the modelling approach. Simulation allows deeper understanding of the system that is being studied, and predicts the performance of different system designs under different sets of input parameters [15].

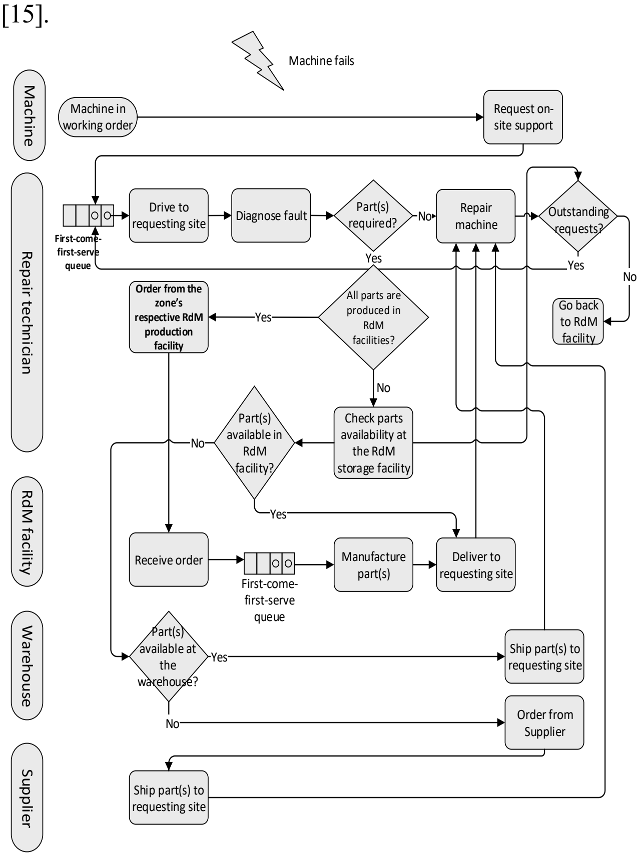

Fig. 1. Generic repair process diagram for the RdM scenario.

Out of the three mainstream simulation methods i.e. discrete-event simulation (DES), system dynamics (SD) and agent-based modelling (ABM); ABM has been chosen as the simulation technique for this study. The reason for choosing ABM could be mainly attributed to the nature of the problem and the nature of the modelled systems. To model individual repair processes as described in the previous section and account for the interaction between the models' constituents and observe the subsequent emerging behavior, an individual-based approach with relatively low level of abstraction is required. This requirement could be achieved through DES or ABM since SD is an abstract high level, top-down modelling approach that models the interaction and feedback loops between aggregates of entities, rather than representing each individual entity on its own [16]. System dynamics have, however, been employed in similar contexts [12,13] as was shown in the Related work section, but its use has been limited to measure inventory and carbon footprint levels only. ABM was preferred over DES since behavior in DES, as Law [17, p.695] puts it, "behaviors are implemented in the model "blocks" that entities pass through, rather than being encapsulated into the entities themselves" as in the case of ABM. Encapsulation is defined by Dennis *et al.,* [18, p.20] as "the combination of process and data into a single entity". ABM is usually implemented through object-oriented programming paradigm [17] as object-orientation, in addition to encapsulation, allows the handling of complexity through the modularization of a system into several smaller simpler individual objects, that are then brought together to form a system [18].

The agent-based models were developed from an object-oriented perspective using AnyLogic, a Java-based simulation development platform. The models consist of five population of agents namely: machines, technicians, suppliers, warehouse, AM equipment (present only in the RdM proposed scenario). Two further agents are also defined in the model to represent non-physical entities, namely: repair Request and zones. Both of these agent types i.e. repair request and zone correspond to Class structure in object-oriented programming and encapsulate data about the failed machine and each zone's allocated resources respectively such as machine location, failure type, required parts and so on. Table 1 below summarizes the model's agents, their key parameters, key functions and key states. Agents marked with asterisk are only present in the RdM scenario, while the rest of the agents exist in both scenarios.

Table 1. Constituents of the agent-based models.

Agent	Key Parameters	Key Functions	Key States
Machine	Number of machines Zone	Process jobs Report failure	Working Failed
Technician	Number of technicians Zone	En-route to machines Inspect machine Order parts Repair	Idle Driving At machine
AM Machine*	Number of machines Zone Rate of production	Producing parts	Idle Producing
Warehouse	Location Inventory level of each part	Order parts replenishments Notify repair personnel of parts availability Deliver parts to RdM's local storage facility	N/A
Supplier	Lead time	Deliver parts to warehouse/ machines	N/A
Repair Request	Machine Failure type	N/A	N/A
Zones*	Assigned AM Assigned machines Assigned technicians		N/A

The communication between agents in both simulation models is performed through messages (data packets) exchange. Messages can carry different types of data and are used to trigger agent actions or to store some information that the agent can later use to make decisions. In addition to messages, agents' actions are also triggered by either deterministic or stochastic timeouts that follow certain probability distributions where an event is triggered after a certain amount of time has passed. Time-out triggered events are mostly used to trigger the completion of tasks such as inspection, repair and delivery processes. Agents' actions in an ABM environment can also be triggered by Boolean conditions, where an agent acts in reaction to the state change within itself, other agent, or within the environment where all agents live. Finally, as agents travel freely in ABM, agents can make decisions or perform some action upon their arrival to a specific location or to some other agent.

Agents behavior and decision making logic is defined and governed through statecharts. Statecharts are modelling constructs based on UML's state machine diagram. Statecharts contain two main elements namely states and transitions. States define the state that the agent is in while the state is active e.g. available or busy, while transitions as discussed above control the moves between the agents' states. States can take two forms namely: simple and composite. A simple state is a standalone state i.e. a state that exists by itself, in other words a state that does not contain further states inside. While a composite state, as the name suggests, is a state that contain further states inside e.g. an agent's state "Busy" could contain further states elaborating exactly what the agent is busy doing such as "Performing job A".

Each statechart has one and only one simple state active at any single moment in time during the simulation run. Each state has two sets of instructions that define agents' actions; one executes immediately upon entering the state, and the other executes once the agent leaves the state. In other words once a state is triggered, its corresponding agent performs some action, then the state stays active i.e. the agent waits for some time to pass or an event to occur, and then once it is time (or condition) to move to another state, the next set of instructions is executed and the agent leaves to the next state. This is best illustrated by an example. Fig. 2. shows the statechart diagram of the technicians, the naming convention of the states follows Java naming conventions. Once the simulation run commences, the technician enters the "atBase" state which means that the technician is idle and ready to head out to attend to a failed machine. Once the technician is assigned a job, through a scheduling algorithm that loops through the failed machines queue and assigns jobs to technicians based on location and the total number of jobs performed by each technician that day. The scheduling rule is applied in the Traditional scenario only, as it is assumed that each zone has only one technician. It could be however used in the RdM scenario, in case more than one technician is assigned to any zone, by looping through each failed machine's zone's allocated technicians. This example is taken from the Traditional scenario.

Entering and leaving the "atMC" state, which refers to "at maintenance center", updates the value of different variables that store time-related information that are used for statistical purposes. Once the scheduling algorithm assigns a job to a technician, it sends the respective technician a message containing information about the job such as the location of the failed machine. Then the technician agent enters the "busy" composite state, which in itself contains two further simple states. Entering the "busy" composite state will execute its corresponding set of instructions, mainly updating the values of a set of variables, then the simple state "enRouteToMachine" becomes active. Entering the "enRoute ToMachine" simple state will instruct the technician agent to move to the location of the failed machine, which was provided in the message received by the technician to attend to the job.

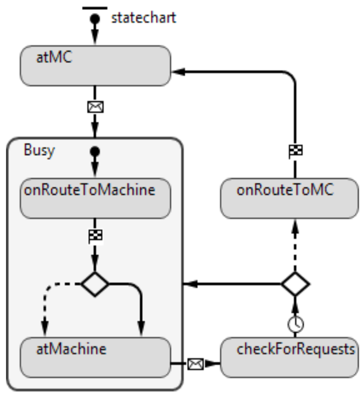

Fig. 2. Technicians' state diagram.

The technician then moves in two dimensional space with a predefined speed (assumed to be 50 miles per hour) to the failed machine. The arrival of the technician to the failed machine triggers the transition between the two simple states "enRouteToMachine" and "atMachine, which has to go through a branch first. Once the technician agent arrives at the failed machine, it communicates with the failed machine agent; so that the corresponding failed machine statechart can be updated accordingly e.g. the failed machine's corresponding active state becomes either "beingInspected" or "beingRepaired" based on whether this is the first visit or the technician has come to finish a job. The technician agent then enters the "atMachine" state and stays there until receiving a message from the failed machine containing information that the repair process has been completed and the machine is back to its operating state, or that there are parts missing and the technician has to order necessary parts and leave. The technician agent then enters the "checkForRequests" simple state, where it checks whether there are any broken machines determined by the scheduling algorithm to be nearby, so it can attend to. If so the technician enters the "busy" composite state and goes through the same steps as above. Otherwise it enters the "onRoteToMC" state where MC is short for maintenance center, the technician then moves in the two-dimensional space back towards the maintenance center, and the "atBase" state becomes active. This was an example of a simple statechart that defines the behavior of the technicians'. During the model runs, transitions between states and interactions between different agents are the driving force behind the behavior of the modelled systems.

The data from Table 2 below that were used as inputs for the model were hypothesized after careful examination of real case studies used in previous research studies and reports from big corporations that are implementing distributed on-demand spare parts production. As noticed from the table, the triangular distribution has been used frequently to model time needed to accomplish tasks. This is mainly because the data used as input were estimates and do not reflect accurate data of an existing system. Indeed the use of the triangular distribution has been suggested in many simulation specialist texts [15,17,19] to model time-related tasks when little or no data are available. The advocacy for the use of the triangular distribution where data are scarce or unavailable is mainly because of the relative ease of estimating its parameters i.e. min, max and mode, which provide a valuable first primal insight about the performance of stochastic systems. The normal distribution was used to trigger the machines failure for the same reason, but high uncertainty, represented by a relatively high standard deviation, and the need to occasionally include extreme values; to better represent random break-downs made the normal distribution a more appropriate choice that is sufficiently generic while maintaining realistic. Since the standard deviation is relatively high (40% of the mean), to avoid getting negative values for the time before failure, the absolute value of the time before failure was used.

Table 2. Key input parameters.

Input parameter	Value
Number of machines	1000
Number of technicians	5
Number of zones	5
Number of failures	10
Number of AM parts	5
Delivery time from central warehouse	1 day
Delivery time from local warehouses	Triangular (3, 6, 8) hours
Delivery from supplier(s)	Triangular (1, 2, 1) days
Machines failure rate	Normal (30, 12) days
Number of AM machines per zone	3
On-demand parts production time	Triangular (8, 12, 10) hours
Annual AM technician salary	£ 60,000
Number of AM machines per technician	3
AM machine price	£ 100,000

5. Results and discussion

Experiments were performed to simulate a service period of five years to evaluate the performance of both scenarios. The evaluation of performance was based on three criteria, namely cost, downtime, and utilization of resources (machines and repair technicians). Table 3 shows an accumulated cost comparison between the two scenarios. It is clear from the cost breakdown below that the implementation of distributed on-demand production of spare parts entails more cost segments (e.g. more personnel, the acquisition of AM machines, AM production cost). Regardless of this, the total cost of this scenario remains lower than the Traditional centralized scenario. It is worth mentioning that the total cost in both scenarios excludes the downtime cost, since this value varies significantly depending on the sector, context, service level agreement and a host of other factors making it difficult to estimate for a generic case.

The main savings from RdM scenario are fairly spread across the entire cost segments spectrum. Although it is expected that the total inventory cost will decrease in the RdM scenario, mainly due to savings on suppliers' orders – as a portion of the parts assortment will be manufactured in-house - and the elimination of a portion of the parts from stock, which means savings on holding costs, the significant decrease in the cost of incurring stock outs and subsequently lower downtimes is high. This decrease in stock outs is mainly due to the assumption that a share of the parts assortment is no longer supplied from a third party supplier, but rather always available after in-house on-demand production and post processing. There was also significant saving on transportation costs since all machines are assigned to zones, each has its own, and shares, dedicated resources. Table 3 below shows a breakdown of the costs for both scenarios.

Table 3. Cost breakdown in £ for both scenarios after 5 years of operations.

Cost	Traditional Scenario	RdM Scenario
Orders cost	16,169,265	12,008,120
Stock out cost	1,964,800	741,400
Holding cost	428,971	339,886
AM production cost	0	2,753,240
Total cost of acquiring AM equipment	0	1,500,000
Total AM technician salaries	0	1,500,000
Total transportation cost	2,468,540	1,187,937
Total cost	21,031,576	20,030,583

It is important however, to note that the cost reduction incurred from the implementation of the proposed RdM scenarios is not apparent on the short-term. It rather takes a few years to begin to observe the cost savings, due mainly to the substantial initial investment required to decentralize operations and acquire advanced production equipment. Fig. 3. below shows a break-even chart for the accumulated cost for both scenarios where the break-even point occurs around the 4th year. This means that unless an OEM is committed to long-term MRO processes, and the depreciation rate of the AM equipment is relatively low, and AM equipment can operate for many years, then the RdM scenario is not appropriate.

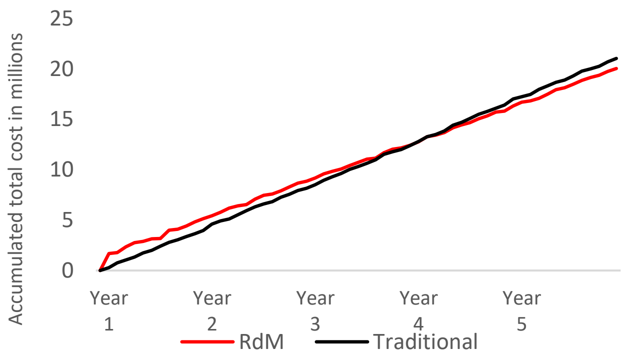

Fig. 3. Break-even chart for the accumulated total cost for both scenarios.

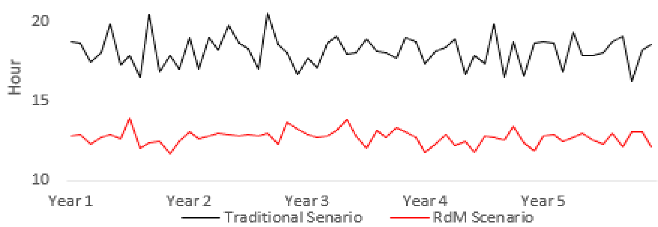

Fig. 4. Average downtimes for both scenarios.

Another decrease was in the utilization of resources. The average availability for all machines (i.e. productive time) in the RdM scenario increased to 98.27 from 97.65 in the Traditional scenario. The availability of the technicians also notably improved in the RdM scenario, as there was an increase of 13% of availability on average in the technicians' available time. This is mainly due to savings on driving times, as technicians only drive in their respective zones. Such availability could be exploited by adding more tasks to the technicians, or increase the number of machine of machines per zones without much compromising the downtime of machines, and consequently associated cost.

The RdM scenario also showed a significant reduction in the average machines' downtime as shown in Fig. 4. above, which could be critical for some industries and could entail some significant penalty for some OEMs. The simulation results showed a decrease of 34.3% in the average downtime of all machines when adopting the RdM scenario. This decrease however, was not solely achieved by the on-demand production of a portion of the spare parts, other factors such as shorter response time due to sharing resources within defined zones, and the existence of small storage facilities near the demand points (in RdM facilities) where the delivery of parts to failed machines is shorter.

6. Conclusion

Simulation modeling was employed to evaluate the performance of two generic repair processes based on total cost, downtime of machines and utilization of resources. The results showed that there is a clear advantage gained from employing RdM into the spare parts sector in all three performance criteria. It was not however the on-demand production that solely contributed to the improved performance, as manufacturing technologies (such as AM equipment) still lack in technical details, are costly to acquire and maintain and could take long time to produce and post process a part. It mainly the distribution of resources, pooling them and allocating them to corresponding zones, and the sharing of resources between agents that live in this zone is what contribute the most to the improvement of the system.

There however remain much to be investigated in this area. First, although insight into generic repair processes was thoroughly developed, it is necessary to evaluate the performance of a case from a particular sector using real data to further validate the model. Second, optimizing parameters is also an important area that is yet to be investigated in this setting. Further, some assumptions were relaxed in order to maintain the simplicity of the model. Such assumptions, when their corresponding data are available, could provide valuable inputs for the model. This research is, however, an important step towards understanding the needs for the adoption of RdM in the production of spare parts, particularly, the decentralization aspects.

7. Reference

1. Chen T., Tsai H. Ubiquitous manufacturing : Current practices, challenges, and opportunities. Robotics and Computer Integrated Manufacturing. Elsevier; 2017; 45: 126–132.
2. Srai JS., Kumar M., Graham G., Phillips W., Tooze J., Ford S., et al. Distributed manufacturing: scope, challenges and opportunities. International Journal of Production Research. Taylor & Francis; 2016; 54(23): 6917–6935.
3. Kohtala C. Addressing sustainability in research on distributed production: An integrated literature review. Journal of Cleaner Production. Elsevier Ltd; 2015; 106: 654–668.
4. Chen D., Heyer S., Ibbotson S., Salonitis K., Steingrímsson JG., Thiede S. Direct digital manufacturing: Definition, evolution, and sustainability implications. Journal of Cleaner Production. 2015; 107: 615–625.
5. Hu Q., Boylan JE., Chen H., Labib A. OR in spare parts management : A review. European Journal of Operational Research. Elsevier B.V.; 2018; 266(2): 395–414.
6. Basten RJI., Houtum GJ Van. Surveys in Operations Research and Management Science System-oriented inventory models for spare parts. Surveys in Operations Research and Management Science. Elsevier Ltd; 2014; 19(1): 34–55.
7. Noyes D. Envisioning e-logistics developments : Making spare parts in situ and on demand State of the art and guidelines for future developments. 2006; 57: 490–503.
8. Holmström J., Partanen J., Tuomi J., Walter M. Rapid manufacturing in the spare parts supply chain. Journal of Manufacturing Technology Management. 2010; 21(6): 687–697.
9. Khajavi SH., Partanen J., Holmström J. Additive manufacturing in the spare parts supply chain. Computers in Industry. 2014; 65(1): 50–63.
10. Liu P., Huang SH., Mokasdar A., Zhou H., Hou L. The impact of additive manufacturing in the aircraft spare parts supply chain: Supply chain operation reference (scor) model based analysis. Production Planning and Control. Taylor & Francis; 2014; 25(13–14): 1169–1181.
11. Khajavi SH., Holmström J. Additive manufacturing in the spare parts supply chain : hub con fi guration and technology maturity. Rapid Prototyping Journal. 2018; 24(7): 1178–1192.
12. Ghadge A., Karantoni G., Chaudhuri A., Srinivasan A. Impact of additive manufacturing on aircraft supply chain performance: A system dynamics approach. Journal of Manufacturing Technology Management. 2018; 29(5): 846–865.
13. Li Y., Jia G., Cheng Y., Hu Y. Additive manufacturing technology in spare parts supply chain: a comparative study. International Journal of Production Research. Taylor & Francis; 2017; 55(5): 1498–1515.
14. Muir M., Haddud A. Additive manufacturing in the mechanical engineering and medical industries spare parts supply chain. Journal of Manufacturing Technology Management. 2017; : JMTM-01-2017-0004.
15. Robinson S. Simulation: the practice of model development and use. Journal of Simulation. 2004. 340 p.
16. Borshchev A., Filippov A. From System Dynamics to Agent Based Modeling: 2004.
17. Law AM. Simulation Modeling and Analysis. Fifth Edit. New York: McGraw-Hill Education; 2015.
18. Alan D., Haley B., Wixom D. Systems Analysis Design with UML Version 2.0: An Object-Oriented Approach. 5th editio. 2015. 525 p.
19. Banks J. Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice. John Wiley & Sons; 1998.

2019-06-24

Redistributed manufacturing of spare parts: an agent-based modelling approach

Haddad, Yousef

Elsevier

Haddad Y, Salonitis K, Emmanouilidis C. Redistributed manufacturing of spare parts: an agent-based modelling approach. Procedia CIRP, Volume 81, 2019, pp. 707-712
https://doi.org/10.1016/j.procir.2019.03.180